Amendments to the Claims

Please amend claims 1, 7, 13, and 19 as follows.

1. (Currently amended) A method of creating programmable data objects for use in a multi-tier computing architecture, the method comprising:

dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing item to a middle-tier stateless programmable data object being created in the visual design surface module, wherein a client process ~~to communicate~~ communicates with the server processing resource through the middle-tier stateless programmable data object using stateless data transfer, wherein stateless data transfer includes receiving a message from the client at the server processing resource that includes:

a first indicia that indicates an original state of data previously requested by the client from the server processing resource; and

a second indicia that indicates changes to the requested data, wherein a current state of the requested data is determined by comparing the first indicia and the second indicia;

identifying data schema associated with the server processing resource added to the middle-tier stateless programmable data object in response to the server processing resource being dropped in the visual design surface module;

creating a typed dataset containing data structures corresponding to the data schema associated with the server processing resource;

creating a command adapter to provide data transfer commands within the middle-tier stateless programmable data object between the middle-tier stateless programmable data object and the server processing resource; and creating a data transfer connection between the middle-tier stateless programmable data object and the server processing resource.

2. (Previously presented) The method according to claim 1, wherein the method further comprises: creating additional initialization methods to support the items added to the middle-tier stateless programmable data object.

3.-6. (Canceled)

7. (Currently amended) A computing system of creating programmable data objects for use in a multi-tier computing architecture, the computing system comprising:

a memory module;

a user interface module;

a mass storage system; and

a programmable processing module, the programmable processing module performing a sequence of operations to implement the following:

dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing item to a middle-tier stateless programmable data object being created in the visual design surface module, wherein a client process ~~to communicate~~ communicates with the server processing resource through the middle-tier stateless programmable data object using stateless data transfer, wherein stateless data transfer includes receiving a message from the client at the server processing resource that includes:

a first indicia that indicates an original state of data previously requested by the client from the server processing resource; and a second indicia that indicates changes to the requested data, wherein a current state of the requested data is determined by comparing the first indicia and the second indicia;

identifying data schema associated with the server processing resource added to the middle-tier stateless programmable data object in response to the server processing resource being dropped in the visual design surface module;

creating a typed dataset containing data structures corresponding to the data schema associated with the server processing resource;

creating a command adapter to provide data transfer commands within the middle-tier stateless programmable data object between the middle-tier stateless programmable data object and the server processing resource; and

creating a data transfer connection between the middle-tier stateless programmable data object and the server processing resource.

8. (Previously presented) The computing system according to claim 7, wherein the sequence of operations further comprises creating additional initialization methods to support the items added to the middle-tier stateless programmable data object.

9.-12. (Canceled)

13. (Currently amended) A computer program product readable by a computing system and encoding instructions for a computing process for creating programmable data objects for use in a multi-tier computing architecture the computing process comprising:

dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing item to a middle-tier stateless programmable data object being created in the visual design surface module, wherein a client process ~~to communicate~~ communicates with the server processing resource through the middle-tier stateless programmable data object using stateless data transfer, wherein stateless data transfer includes receiving a message from the client at the server processing resource that includes:

a first indicia that indicates an original state of data previously requested by the client from the server processing resource; and

a second indicia that indicates changes to the requested data, wherein a current state of the requested data is determined by comparing the first indicia and the second indicia;

identifying data schema associated with the server processing resource added to the middle-tier stateless programmable data object in response to the server processing resource being dropped in the visual design surface module;

creating a typed dataset containing data structures corresponding to the data schema associated with the server processing resource;

creating a command adapter to provide data transfer commands within the middle-tier stateless programmable data object between the middle-tier stateless programmable data object and the server processing resource; and

creating a data transfer connection between the middle-tier stateless programmable data object and the server processing resource.

14. (Previously presented) The computer program product according to claim 13, wherein the computing process further comprises creating additional initialization methods to support the items added to the middle-tier stateless programmable data object.

15.-18. (Canceled)

19. (Currently amended) A system comprising a processor executing instructions for creating programmable data objects for use in a multi-tier computing architecture, the system comprising:

a server explorer module for presenting one or more server resources present on a server to a programmer for use in creating a programming object class; and

a visual design surface module for performing the operations associated with creating, editing, and saving a middle-tier stateless programmable data object, the visual design surface module comprising:

a drag/drop module for enabling a programmer to perform a drag and drop of a server resource onto the visual design surface module, the drag and drop including selecting a server resource from the server explorer module and placing the selected server resource within the middle-tier stateless programmable data object on the visual design surface module, wherein a client process ~~to communicate~~ communicates with the server resource through the middle-tier stateless programmable data object using stateless data transfer, wherein stateless data transfer includes receiving a message from the client at the server resource that includes:

a first indicia that indicates an original state of data previously requested by the client from the server resource; and

a second indicia that indicates changes to the requested data, wherein a current state of the requested data is determined by comparing the first indicia and the second indicia;

a command adapter function generation module for generating a command adapter module associated with the drag and drop of the server resource;

a typed dataset generation module for generating a typed dataset object associated with the drag and drop of the server resource;

an init generation module for generating processing functions and methods of a data transfer connection between the middle-tier stateless programmable data object and the server resource associated with the drag and drop of the server resource; and

a properties edit module for retrieving properties and source code for objects within the visual design surface module for editing.

20. (Previously presented) The system according to claim 19, wherein the drag/drop module comprises:

an explorer interface module to select the server resource from the server explorer module and place it within the middle-tier stateless programmable data object within the visual design surface module;

a user interface module to perform the visual display and command input operations associated with the drag and drop; and

a class generation module to cause the visual design surface module to perform the operations to complete the drag and drop of the server resource onto the visual design surface module.

21. (Previously presented) The system according to claim 19, wherein the drag/drop module further causes the other processing modules in the visual design surface module to perform their operations to complete the drag and drop of the server resource onto the visual design surface module.

22. (Previously presented) The system according to claim 19, wherein the command adapter function generation module comprises:

a GetDS module for generating a GetDataSet function that fills a typed dataset with data obtained from the server resource; and

an updateDS module for generating an UpdateDataSet function that updates the server resource using the data stored within the typed dataset.

23. (Previously presented) The system according to claim 19, wherein the command adapter function generation module further accepts an updated command adapter module that has been edited by the properties edit module and generates updated source code for functions within the command adapter module.

24. (Previously presented) The system according to claim 19, wherein the typed dataset generation module comprises:

a Table Schema module for generating the table records from a database schema within the typed dataset object;

a Relations module for generating the relationship data for the fields within the records within the typed dataset object based upon the corresponding relationship data from the server resource; and

a Views module for generating the database views data for the records within the typed dataset object based upon the corresponding views data from the server resource.

25. (Previously presented) The system according to claim 19, wherein the typed dataset generation module further accepts an updated typed dataset module that has been edited by the properties edit module and generates updated source code for functions within the typed dataset object.

26. (Previously presented) The system according to claim 19, wherein the init generation module comprises:

an Init Function module for generating the processing functions and methods within the middle-tier stateless programmable data object associated with the command adapter modules;

an InitDataSet module for generating the processing functions and methods within the middle-tier stateless programmable data object module associated with the type dataset class; and

an InitConnection module for generating the processing functions and methods within the middle-tier stateless programmable data object associated with the data transfer connection between the middle-tier stateless programmable data object and the server resource.

27. (Previously presented) The system according to claim 19, wherein the init generation module further accepts an updated Code Generated Method module that has been edited by the properties edit module and generates updated source code for the functions and methods of the data transfer connection.

28.-30. (Canceled)

31. (Previously presented) The method according to claim 1, further comprising:

inserting a database connection module that creates the data transfer connection between the middle-tier stateless programmable data object and the server processing resource when the dragged item is a database table within the server processing resource, wherein the database connection module comprises:

a data connection object for creating and managing the data transfer connection between the middle-tier stateless programmable data object and the server processing resource;

a managed resource module for providing the data connection object with address and identification information to establish the data transfer connection; and

a persistent data storage for maintaining the address and identification information used by the managed resource module.

32. (Previously presented) The computing system according to claim 7, wherein the programmable processing module further performs the following:

inserting a database connection module that creates the data transfer connection between the middle-tier stateless programmable data object and the server processing resource when the dragged item is a database table within the server processing resource, wherein the database connection module comprises:

a data connection object for creating and managing the data transfer connection between the middle-tier stateless programmable data object and the server processing resource;

a managed resource module for providing the data connection object with address and identification information to establish the data transfer connection; and

a persistent data storage for maintaining the address and identification information used by the managed resource module.

33. (Previously presented) The computer program product according to claim 13, wherein the computing process further comprises:

inserting a database connection module that creates the data transfer connection between the middle-tier stateless programmable data object and the server processing resource when the dragged item is a database table within the server processing resource, wherein the database connection module comprises:

a data connection object for creating and managing the data transfer connection between the middle-tier stateless programmable data object and the server processing resource;

a managed resource module for providing the data connection object with address and identification information to establish the data transfer connection; and

a persistent data storage for maintaining the address and identification information used by the managed resource module.

34. (Previously presented) The system according to claim 19, wherein the properties edit module further comprises:

a Properties Retrieval module for retrieving the properties and source code for the objects within the visual design surface module for editing;

a User Interface module for presenting the properties and source code to a programmer, and accepting edits from the programmer; and

a Class Update module for identifying all other items within the visual design surface module that are affected by the changes made to the edited item and updating the identified items consistent with the edits made to the edited item.

35. (Previously presented) The method according to claim 1, further comprising:

editing the processing items within the visual design surface module, wherein the processing items comprise properties and processing instruction source code;

updating the processing items edited within the visual design surface module;

updating the typed dataset within the visual design surface module;

updating the command adapters within the visual design surface module;

identifying any other processing items containing references to the data structures and functions edited; and

updating the identified items containing references to data structures and functions edited to make all references consistent with each other.

36. (Previously presented) The computing system of claim 7 wherein the sequence of operations further comprises:

editing the processing items within the visual design surface module, wherein the processing items comprise properties and processing instruction source code;

updating the processing items edited within the visual design surface module;

updating the typed dataset within the visual design surface module;

updating the command adapters within the visual design surface module;

identifying any other processing items containing references to data structures and functions edited; and

updating the identified items containing references to the data structures and functions edited to make all references consistent with each other.

37. (Previously presented) The computer program product according to claim 13, wherein the computing process further comprises:

editing the processing items within the visual design surface module, wherein the processing items comprise properties and processing instruction source code;

updating the processing items edited within the visual design surface module;

updating the typed dataset within the visual design surface module;

updating the command adapters within the visual design surface module;

identifying any other processing items containing references to data structures and functions edited; and

updating the identified items containing references to the data structures and functions edited to make all references consistent with each other.

38. (Previously presented) The system according to claim 19, further comprising a database connection module for creating the data transfer connection between the middle-tier stateless programmable data object and the server resource, wherein the server resource includes a database table, the database connection module including:

a data connection object for creating and managing the data transfer connection between the middle-tier stateless programmable data object and the server resource;

a managed resource module for providing the data connection object with address and identification information to establish the data transfer connection; and

a persistent data storage for maintaining the address and identification information used by the managed resource module.